REMARKS

Claims 1-16 of the application stand rejected. Claims 1, 2, 5, 8, 10, 12, 14 and 15 have been amended herein to more clearly define the scope of the presently claimed invention. Applicant respectfully requests reconsideration of pending Claims 1-16 in light of the amendments and remarks herein.

35 U.S.C. §103

Claims 1, 7, 8 and 10 stand rejected under 35 U.S.C. §103 as being unpatentable over U.S. Patent No. 5,754,862 ("Jones") in view of U.S. Patent No. 6,016,392 ("Jordan"). Applicant respectfully traverses the rejection. Most importantly, Applicant submits that the combination of Jones and Jordan is improper as it relies on hindsight. Jones teaches a method and system for implementing object-oriented techniques of virtual functions and virtual classes (Jones, Col. 5, lines 16-19). The teaching in Jordan, on the other hand, focuses on allocation of memory in a computer in order to represent objects supported in an object-oriented programming environment. Applicant submits that simply because both references generally address issues in an object-oriented environment does not render the combination of the references proper. In fact, Applicant respectfully contends that barring hindsight, there is no teaching in either reference to suggest that these references may be properly combined and/or that one of ordinary skill in the art would think to combine the references. Applicant therefore respectfully submits that the combination of these references is improper.

Even assuming arguendo these references were properly combined, Jones and/or Jordan do not teach or suggest the invention in Claims 1, 7, 8 and 10. As described in the specification, in embodiments of the invention, the internal data structure of objects which are instances of a class that implements interfaces may be modified to include extra fields, where the extra fields include pointers to interface vtables for the interfaces implemented by the class. These pointers may allow more efficient dispatch of interface functions and/or allow the efficient casting of references of an interface type into references whose type is defined by the class that implements the interface

(Specification, Page 9, lines 11- 18). These features are explicitly claimed in independent Claims 1, 7, 8 and 10.

Jones does not teach or suggest the use of pointers in this manner. The sections the Examiner highlights in Jones merely describe typical object oriented programming elements, including functions, classes, objects and/or interfaces. There is no teaching or suggestion in these sections (or any other sections of Jones) that the elements be combined in the manner claimed. Specifically, there is no teaching or suggestion of an interface vtable comprising a first pointer configured to point to a function and/or an object that is an instance of a class, where the object comprises a second pointer configured to point to the interface vtable associated with an interface. The combination of Jordan with Jones also does not teach or suggest these elements of Claim 1.

Similarly, Claims 8 and 10 include the elements of receiving a first pointer configured to point to an interface vtable associated with an interface, an object comprising the first pointer, the object being an instance of the class that implements the interface, and receiving a second pointer configured to point to the function, the interface vtable comprising the second pointer. Jones, alone and/or in combination with Jordan does not teach these elements. Jones and Jordan therefore do not render independent Claims 1, 8 and 10 unpatentable, and as a result, also do not render dependant Claim 7 unpatentable.

Claims 2, 3 and 4 stand rejected under 35 U.S.C. §103 as being unpatentable over Jones in view of Jordan, in further view of AP ("Arrays, pointers, pointer arithmetic"). Claims 5 and 6 stand rejected under 35 U.S.C. §103 as being unpatentable over Jones in view of Jordan, in further view of Kathleen Fisher, et al. ("What is an Object Oriented Programming Language?", hereafter "Kathleen"). Claims 9 and 11 stand rejected under 35 U.S.C. §103 as being unpatentable over Jones in view of Jordan, in further view of Danel Liang ("Java Programming"). Applicant respectfully traverses these rejections.

Claims 2-6 are dependant on independent Claim 1, Claim 9 is dependant on Claim 8 and Claim 11 is dependant on Claim 10. As previously described, Jones, alone or in combination with Jordan, does not render independent Claims 1, 8 and 10 unpatentable. The addition of AP, Kathleen and/or Java Programming to Jones and/or Jordan also does

not teach or suggest the elements of Claims 1, 8 and/or 10, and therefore these references also do not render the dependant Claims 2-6, 9 and 11 unpatentable.

Claims 12, 13, 15 and 16 stand rejected under 35 U.S.C. §103 as being unpatentable over TO ("Object Reference Casting") in view of AP. Applicant respectfully traverses the rejection. TO includes a discussion of object reference casting (TO, Page 1) while AP appears to be a discussion of how arrays are accessed in C (AP, Page 1). These references thus appear to discuss abstract programming concepts, without any suggestion that the concepts may be combined to achieve the claimed invention. More specifically, the references, alone and/or in combination, do not teach or suggest the invention in Claims 12, 13, 15 and 16. Claims 12 and 15 are independent claims directed to a *novel* method for casting a reference to an object, according to the claimed elements. The claims are not directed at the *general* concept of casting, as discussed in TO. Indeed, the general concept of casting, as discussed in TO is also described in the Background section of the application ("A programming language or computing environment may provide an instruction to convert or cast a reference of one type to another type, e.g., a reference of a type defined by a class may be cast into a reference of a type defined by an interface that is implemented by the class." Specification, Page 3, lines 1-4).

Embodiments of the invention as claimed, however, include receiving a first reference configured to refer to an object, the first reference having a type defined by an interface, receiving a request to cast the first reference to a type defined by a class that implements the interface, and receiving a pointer, the pointer contained in the object, the pointer configured to point to a canonical base address of the object. The Examiner concedes that TO does not teach the element of receiving a pointer, the pointer contained in the object, the pointer configured to point to a canonical base address of the object. The Examiner, however, submits that AP teaches this element. Applicant strongly disagrees. AP merely describes the use of pointers to access arrays (a sequence of objects). There is no teaching or suggestion in AP for the use of pointers in a method to cast a reference to an object. Again, Applicant reiterates that TO and AP may not be blindly combined to suggest that the concept of using pointers, as claimed, to cast a reference to an object is known. On the contrary, since there is no suggestion in TO for

the use of a pointer contained in an object, where the pointer is configured to point to a canonical base address of the object, and AP also does not teach this element, Applicant submits that the combination of TO and AP does not teach or suggest the claimed invention in Claims 12 and 15. Claims 13 and 16 are dependant on Claims 12 and 15 respectively. TO and AP therefore also do not render these claims unpatentable.

Finally, Claim 14 stands rejected under 35 U.S.C. §103 as being unpatentable over TO in view of AP, in further view of U.S. Patent No. 6,421,681 B1 ("Gartner"). Claim 14 is dependant on Claim 12. As previously discussed, TO and AP do not teach or suggest the elements of Claim 12. The addition of Gartner to TO and AP also does not teach or suggest these elements of Claim 12. Since Claim 14 is dependant on Claim 12, TO, AP and/or Gartner therefore also do not render Claim 14 unpatentable.

In summary, Applicant respectfully submits that none of the references cited, alone and/or in combination, render Claims 1-16 unpatentable. Applicant therefore respectfully requests the Examiner to withdraw the rejection to Claims 1-16 under 35 U.S.C. §103.

CONCLUSION

Based on the foregoing, Applicant respectfully submits that the applicable objections and rejections have been overcome and that pending Claims 1-16 are in condition for allowance. Applicant therefore respectfully requests an early issuance of a Notice of Allowance in this case. If the Examiner has any questions, the Examiner is invited to contact the undersigned at (310) 406-2362.

Respectfully submitted,

Dated: April 8, 2004

Sharmini N. Green
Senior Attorney
Intel Corporation
Registration No. 41,410
(310) 406-2362

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025
(310) 207-3800

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to: Mail Stop Fee Amendment, Commissioner for Patents, Post Office Box 1450, Alexandria, Virginia 22313-1450 on April 8, 2004.

Margaux Rodriguez *April 8, 2004*